systems

MDPI

Article

Research on Financial Fraud Detection Models Integrating Multiple Relational Graphs

Jianfeng Li * and Dexiang Yang

School of Economics and Management, China Jiliang University, Hangzhou 310018, China; s20071201034@cjlu.edu.cn
* Correspondence: jianfengli@cjlu.edu.cn

Abstract: The current fraud risk in digital finance is increasing year by year, and the mainstream solutions rely on the inherent characteristics of users, which makes it difficult to explain fraud behaviors and fraud behavior patterns are less researched. To address these problems, we propose an integrated multiple relational graphs fraud detection model Tri-RGCN-XGBoost, which analyzes the impact of user association patterns on fraud detection by mining the behavioral associations of users. The model builds a heterogeneous information network based on real transaction data, abstracts three types of bipartite graphs (user–device, user–merchant, and user–address), aggregates the information of the user's neighbor nodes under the three types of behavioral patterns, and integrates the graph convolution classification results under the three behavioral patterns with the XGBoost model to achieve fraudulent user detection with integrated multiple relational graphs. The results show that the performance of this model in fraud identification is significantly improved, especially in reducing the fraudulent user underreporting rate. Further, the behavioral associations that play a key role in fraud user identification are analyzed in conjunction with shape value to provide a reference for fraud pattern mining.

Keywords: fraud detection; graph representation learning; heterogeneous information networks; decision integration

Citation: Li, J.; Yang, D. Research on Financial Fraud Detection Models Integrating Multiple Relational Graphs. *Systems* **2023**, *11*, 539. https://doi.org/10.3390/systems11110539

Academic Editor: William T. Scherer

Received: 11 September 2023
Revised: 28 October 2023
Accepted: 30 October 2023
Published: 4 November 2023

1. Introduction

Nowadays, digital finance is booming under the dual-wheel drive of financial data and digital technology, providing new ways and means to serve the real economy and resolve financial risks. Along with this, the "black industry" has also taken root in the shadows, penetrating the fields of personal credit, e-commerce, and various types of insurance. The report released by the Security Attack and Defense Laboratory of the Industrial and Commercial Bank of China (ICBC) showed that fraud against digital finance has caused losses of USD 174 billion in 2021, and has developed into a well-organized black industrial chain with a clear division of labor. Financial marketing scenarios, for example, platforms or merchants to enhance competitiveness, to attract users to provide red packets, cashback, full reduction, and other marketing activities, have given rise to several marketing fraud gangs focused on the above marketing activities to obtain economic benefits [1]. In recent years, this kind of fraud has shown a trend of "professionalization, specialization, teamwork, and transnationalization", which has caused serious losses to both customers and financial institutions. Therefore, it is of great theoretical value and practical significance to explore financial anti-fraud methods of in new scenarios.

Mainstream fraud detection methods can be categorized into three types: rule-based, machine learning-based, and graph representation learning-based. Rule-based strategies rely on expert experience to design strong rules for risk scoring, but they do not apply to current fraud detection scenarios due to the increased covertness of defrauding users through tampering with IPs and other means [2]. In contrast, data-driven machine learning

methods show better performance in complex tasks, but machine learning also faces certain challenges around user feature representation learning. The most immediate problem is feature ambiguity, where fraudulent users can hide their identities by maliciously mimicking and packaging them as normal transactions, which largely affects model recognition accuracy. The deeper problem is that, for the trend of fraud ganging, the traditional feature mining perspective of machine learning will ignore the information of the fraud gang interaction network and ignore the association information between entities [3]. Based on this idea and thanks to the natural information representation ability of graph structures, graph representation learning has gradually been applied to fraud detection research [4]. Traditional graph representation learning focuses on quantifying the degree of association between nodes, and fraud detection is achieved through subgraph partitioning, which is commonly used (for example by Fraudar [5]), and community discovery algorithms [6,7]. However, traditional graph representation learning exists only for node relationship modeling, and the existence of node features can take full advantage of the disadvantages of the information, graph neural network algorithm well-integrated node relationships, and node features of the two types of information, resulting in a better performance in financial fraud detection [8]. Yu et al. proposed the message-passing-based graph convolution network (MP-GCN) to detect phishing frauds, which is based on a graph convolution neural network to achieve a transaction network graph embedding which compresses the node network into a low-dimensional embedded representation and learns to reconstruct the topological information of the network and the features of the nodes [9]. Jing et al. proposed a method of learning parametric adjacency matrices, which relies on the similarity of the features to deliver messages, to improve the GCN layer to extract node features, and to effectively improve the performance of credit card fraud detection with a small number of samples [10]. In addition to the utilization of node information, Zhu et al. considered the influence of external factors and designed an Attribute Enhanced Spatio-Temporal Graph Convolutional Network (AST-GCN) to encode these factors and integrate them into the spatio-temporal graph convolutional model to improve the model detection accuracy [11].

All of the above studies are for isomorphic diagrams, but in real environments, the diagram structure of the composition between things is often heteromorphic. In this regard, Kanezashi et al. evaluated representative homogeneous GNN models and heterogeneous GNN models and emphasized that the diversity of semantic relationships of heteromorphic graph nodes can effectively improve model fraud detection performance [12]. Currently, in heterogeneous information network fraud detection, the main research method used is to obtain the potential expression of nodes by synthesizing node information and network topologies. Meanwhile, there are fewer studies on the impact of different association relationships in heterogeneous graph fraud detection, and fewer studies focusing on different association behavior patterns to explain the feasibility of fraudulent behaviors.

Combining the above considerations, this paper proposes Tri-RGCN-XGBoost, a multiple relational heterogeneous graph neural network fraud detection algorithm, to capture the intrinsic patterns of fraudulent users or their interaction behaviors. The specific approach transforms the user behavior data into a heterogeneous network, abstract the user–device, user–merchant, and user–address relationship graphs, use the graph neural network model to aggregate the three kinds of meta-paths of the fraudulent user's interaction patterns. These are combined with the XGBoost tree model to complete the fusion of decision-making under the three kinds of relationship graphs, to achieve fraudulent user detection. The experimental comparison of this paper's model with the baseline model on four real enterprise datasets show a significantly improvement in evaluation indexes such as recall rate, proving the accuracy and effectiveness of this model. In addition, the key association behaviors are further analyzed by feature importance ranking, which highlights the effective enhancement of targeted association behavior mining on fraud detection in terms of relationship graph importance.

2. Literature Review

2.1. Fraud Detection Based on Graphs

As a special data structure, graph data can explicitly represent complex relationships between data, which can assist the analysis process, mine the structural information behind the data, and provide more accurate and convincing decision support. Graph Neural Networks are widely used in areas such as financial fraud, audit fraud, and fake news fraud detection [13]. With the gradual complexity of application scenarios, the demand for classification detection tasks for heterogeneous graphs in multiple relational scenarios is gradually increasing. The optimization of heterogeneous graph neural network models mainly focuses on the optimization of information aggregation strategies on the one hand, and the optimization of the architecture of graph convolutional neural networks on the other hand. Jiang et al. investigated the MAFI model by applying multiple types of graph convolutional aggregators to different relationships to achieve aggregated neighbor information and vector implicit and explicit feature interaction updates, and using aggregator-level attention to learn the importance of different aggregators [14]. The mining of fraudulent behavior patterns has some limitations in the real environment where fraudulent behaviors are complex and changeable; Zhang et al. studied the normal patterns, building a competitive graph neural network to model normal and fraudulent behaviors separately, and the normal behavior profiles built by directing the construction of some normal behaviors as weakly regulated information to weaken the dependence on fraudulent behavior patterns [15]. On the representation of topological network structure, Tan et al. studied the main practices of fraud detection techniques for heterogeneous information networks and concluded that only considering the shallow topology of the network is not favorable for capturing the nonlinear relationship between nodes [16]. On the other hand, some studies considered the interference of data imbalance on fraud detection and proposed strategies such as subgraph classification. Li et al. improved the model for information aggregation means to expand the differences between heterogeneous neighbor nodes by increasing the similarity of homogeneous neighbors in the balanced neighborhood [17]. Liu et al. [18] constructed the Pick and Choose Graph Neural Network (PC-GNN) model by combining subgraph filtering in the face of the problem of severe skewing of data labels in heterogeneous graphs. A label balancing sampler is designed to select nodes and edges to construct subgraphs for small batch training to select neighboring candidate nodes, followed by graph convolution to aggregate information from selected neighbors and different relationships [18]. Zhang et al. analyzed key behavioral patterns in spam detection based on business considerations and integrated multiple subgraphs to enhance the representation of comment information in correlation with aggregation operations [19].

2.2. Graph-Based Feature Mining

As the level of anti-fraud technology continues to improve, the frauds that can be committed by a single individual are becoming more and more difficult, and thus, fraud gangs have begun to operate in an organized manner. As an analytical tool to reveal the natural relationship network, a graph can accurately reflect the close association between the fraudster and the fraudulent behavior, and identify identity forgery, identity impersonation, proxy packaging, group fraud, and other malicious behaviors from normal behaviors, which is more advantageous. Some scholars have researched mining graph structure features combined with machine learning models to directly enhance the classification and recognition ability of fraud detection models with graph features. Such studies generally choose to obtain graph structure features of static graphs, which include features such as node degrees and node similarity. Hassanzadeh et al. [20] collected structural features such as centrality and community cohesion on social networks and added them to the model to detect abnormal users [20]. Fu et al. used user communication data to construct a relational network, combined with the DeepWalk algorithm to extract the graph features of user nodes and input them into a LightGBM model; the experiments proved that the structural features improved significantly compared to the user's intrinsic feature fraud

detection effect [21]. Other research utilizes community-based anomaly detection methods to find nodes that are closely connected in the graph, e.g., Moradi et al. detect anomalies based on the idea that anomalous nodes are active in more than one community by using a community detection algorithm to discover communities that violate community boundary rules [22]. Qian et al. used the community matching method for network anomaly detection to associate communities in adjacent time steps and detected anomalies by monitoring the changes in the characteristics of each community to achieve optimization of community discovery on dynamic data [23].

In general, at present, most of the financial transaction fraud detection methods tend to rely on the risk data and features obtained from the downstream transaction side or individual node data and usually take a single user as the detection target. Meanwhile, in contrast, most of the feature systems are built mainly based on the fixed attributes of users, but there is an obvious lack of correlation between information mining and gang behavior. Since there are also serious deficiencies in the detection of fraudulent gang behavior in financial transactions and the interpretability of models, the rational use of graph data tools for visual information integration, the introduction of graph features of correlated information, and the fusion of correlated information and data knowledge can achieve more efficient fraud detection.

3. Multiple Relational Graphs Neural Network Fraud Detection

3.1. Basic Definition

In financial transaction activities, users logging in, operating, and purchasing goods will leave certain information records, which cover information related to users, devices, merchants, and address entities, and their relationships are shown in Figure 1. For easy understanding, we made the following definitions.

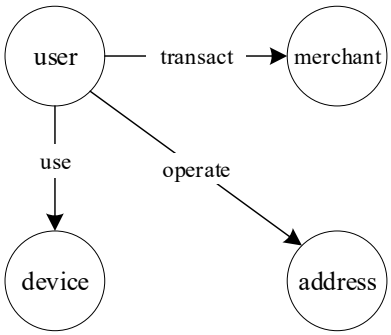


Figure 1. Heterogeneous information network.

Definition 1. *Heterogeneous information graph: To further analyze the fraud pattern of fraudulent users so as to more intuitively mine the behavioral trajectory of fraudulent users, the transaction data are transformed into a heterogeneous information network. The directed graph information network consists of a set of nodes V and a set of edges E.* $G = (V, E; \varphi, \gamma; A, B)$*, where the node mapping function is* $\varphi : V \to A$*, and the connection mapping function is* $\gamma : E \to B$*. They hold the types of nodes and edges, respectively, and can store different types of nodes and connections. In the user heterogeneous information graph, the node set V includes four types of nodes, namely, users, devices, merchants, and addresses.*

Definition 2. *Relationship graph: To discuss the influence of the relationship between users and other types of nodes on the fraud detection effect, the three kinds of relationships appearing in the user heterogeneous information graph are split to obtain a single relationship graph. In this paper, we focus on the user–device, user–merchant, and user–address connections. Given* $G^{(i)} = (X, Z^{(i)}, E; F)(i = 1, 2, 3)$*, where the user node* $X = \{x_1, x_2, \cdots, x_n\}$ *denotes the n-bit user and F holds the user characteristics and* $Z^{(i)} = \{z_{i1}, z_{i2}, \cdots, z_{im}\}(i = 1, 2, 3)$ *denotes the three types of nodes (device, merchant, and address),* $E^{(i)} = \left\{ e_{kj}^{(i)} \right\}_{j=1,2,\cdots,m}^{k=1,2,\cdots,n}$ *preserves the set of*

directed edges from X to Z. The different bipartite graphs in the fraud situation are the different perspectives associated with the multiple relational graphs. The three types of relationship graphs, user–device, user–merchant, and user–address, are denoted as GUD, GUM, and GUA, respectively.

3.2. Model Structure

The model of this paper is shown in Figure 2. We construct the Tri-RGCN-XGBoost model which mainly consists of two parts: the first part is Tri-RGCN and the second part is the XGBoost layer. RGCN (relational GCN) is a model evolved based on a graph convolutional neural network to solve the heterogeneous graph problem, which is based on the aggregation of neighboring nodes from the local structure, and the encoded features are very effective for the task of supervised node classification. The Tri-RGCN layer uses three independent RGCN models to achieve the classification task and outputs the classification results of the three classes of relational graphs.

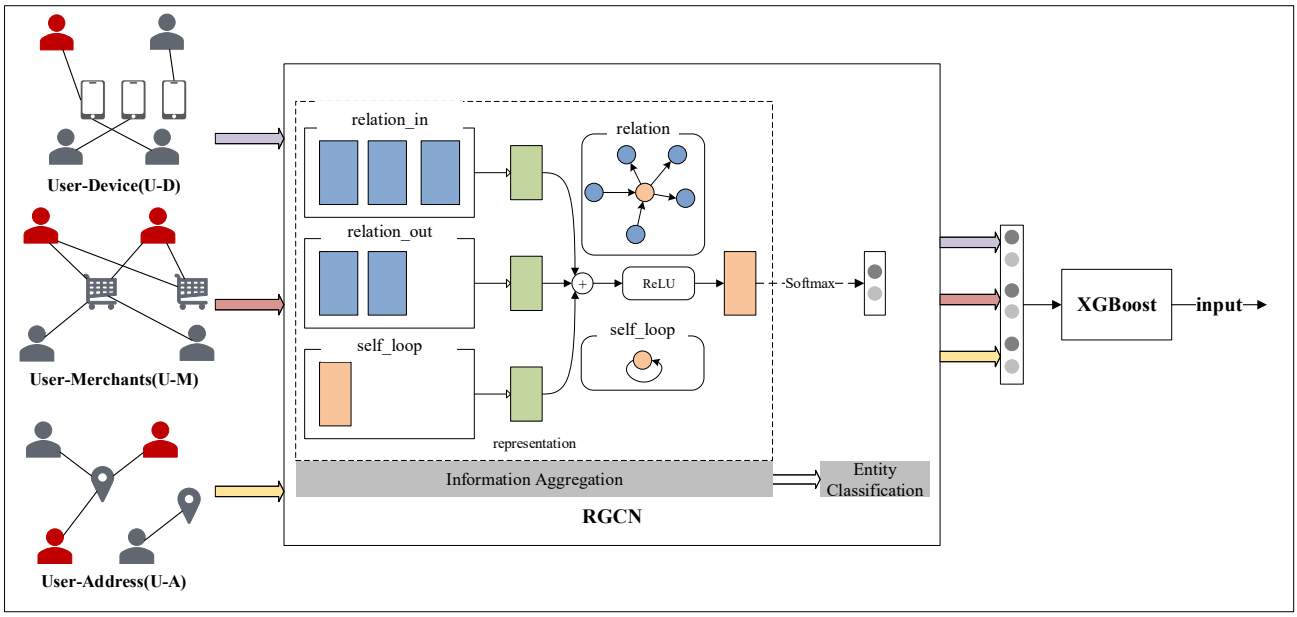


Figure 2. Tri-RGCN-XGBoost model.

The second part is the information integration phase, which fuses the classification results of the three types of behavioral patterns through the tree model XGBoost, taking into account the variability of the three types of relationships in terms of fraud identification effects after information aggregation.

3.2.1. RGCN Layer

RGCN (relational GCN) is an evolution of Graph Convolutional Networks for heterogeneous networks, the core of which is the use of a graph convolutional model to deal with the transfer of multiple types of relational interaction information in the network structure. In addition to the advantages of the RGCN model itself in the representation of node information, Huang et al. designed experiments to significantly improve the performance of RGCN compared to the GCN model with equal inputs of the same heterogeneous graph [24]. The RGCN model is divided into encoder and decoder modules, where the encoder link combines the node representation with the edge types to compute the updated weight matrix, aggregates the input information, and generates a new node representation. The node information aggregation update formula is as follows:

$$h_u^{(l+1)} = \sigma\left(\sum_{i\in E^{(i)}}\sum_{j\in N_u^i}\frac{1}{c_{u,i}}W_i^{(l)}h_j^{(l)} + W_0^{(l)}h_u^{(l)}\right) \quad (1)$$

In the activation function named σ, N_u^i denotes the set of neighboring nodes of node x_u with relation i, $c_{u,i} = \left|N_u^i\right|$ is a regularization constant, $W_i^{(l)}$ is a linear transformation function used to deal with the same type of edge neighboring nodes, $h_u^{(l)}$ is the lth layer node representation of the node x_u, and $h_j^{(l)}$ denotes the neighboring node representation of the node under relation i in the lth layer of the node.

The decoder module performs node category prediction using the softmax activation function on the updated node vector representation, combines some of the node labels, and supervises the learning of model parameters by minimizing the cross-entropy loss function:

$$L = -\sum_{u \in X} \sum_{k=1}^{K} y_{uk} \ln h_{uk}^{(L)} \tag{2}$$

where X is the set of node indices that have labels, $h_{uk}^{(L)}$ is the k-th entry of the network output for the u-th labeled node, and y is the real label of the node.

The RGCN model for a bipartite graph only needs to consider one relationship, which effectively alleviates the overfitting problem caused by parameter proliferation due to too many relationships. The Tri-RGCN layer consists of three independent RGCN models, and the RGCN model realizes the corresponding relationship information transfer and node classification on the bipartite graphs GUD, GUM, and GUA, and outputs three independent classification results.

3.2.2. XGBoost Layer

XGBoost (eXtreme Gradient Boosting) is an integrated learning algorithm based on GBDT. Integrated learning is a framework in which several base classifiers are combined to form strong classifiers to complement each other's strengths and weaknesses, which is used to reduce the error of a single model in order to improve the overall robustness [25]. XGBoost generates a weak learner by optimizing the structured loss function, directly utilizes the values of the first-order and second-order derivatives of the loss function, searches for optimal splitting points of the tree model through the derivatives, and uses techniques such as pre-sorting, weighted quartiles, and so on, to greatly improve the performance of the algorithm. The regular term is added to the objective function to control the complexity of the model to prevent overfitting, which is more flexible than the GBDT model, and the accuracy is improved by using the second-order derivatives.

3.3. Evaluation Metrics

The experiment belongs to a typical binary classification problem with unbalanced data, and the classification effect of the fraud detection model for positive class (fraudulent user-1) and negative class (normal user-0) uses a confusion matrix combined with ROC curve and AUC as evaluation metrics. Firstly, the prediction results can be classified into four categories:

(1) *TP* (true example): The account is predicted to be a fraud account with a true label of fraud.
(2) *FP* (false positive case): The account is predicted to be fraudulent and its true label is normal.
(3) *FN* (true counterexample): The account is predicted to be normal and its true label is fraud.
(4) *TN* (false counterexample): The account is predicted to be normal, and its true label is normal.

Based on the confusion matrix, Accuracy (*Acc*), Precision (*P*), and Recall were used as evaluation metrics. Accuracy refers to the proportion of all users whose prediction labels are correct, Precision indicates the proportion of fraudulent users that contain real fraudulent

users in the prediction results, and Recall indicates the proportion of real fraudulent users that are predicted accurately. They are calculated with the following equations:

$$Acc = \frac{TP + TN}{TN + TP + FN + FP} \tag{3}$$

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

Since it is difficult to achieve good results in both accuracy and recall in the prediction process, $F1$_score is used to reconcile and average the two to evaluate the model more comprehensively, and the formula is:

$$F1\_score = \frac{2P \cdot R}{P + R} \tag{6}$$

4. Experiment and Results

4.1. Experimental Dataset

The data source is the China DataCastle Competition and 2018 Sweet Orange Financial Cup Big Data Modeling Competition; the original data set includes 31,179 users, 1.46 million operational data points, and 260,000 transaction data points, corresponding to the labeled data, i.e., the target variable. The data time span is 30 days, and the number of users labeled as fraudulent is 4285, which accounts for 13.7% of the total number of users. The operation and transaction data are all user behavior data, containing 17 and 26 columns of raw fields, respectively, which can be roughly divided into four categories of raw fields: device information, operation information, transaction information, and merchant information; some variables were introduced, as shown in Table 1.

Table 1. Some of the original variables.

Category	Name	Description
Device	device1–device2	Device parameters
	device_code1–device_code3	Device unique identification
	ip1–ip2	Device IP address
	geo_code	Geographic location code
Operation	mode	Operation type
	day, time	Operation time
Transaction	channel	Transaction platform
	trans_amt	Transaction amount
Merchant	merchant	Merchant identification
	acc_id1	Transaction account code
	acc_id2	Transfer in account code

4.1.1. Feature Mining

The default values of the raw data were processed according to the requirements of feature mining. First, the raw variable features were summarized and analyzed, and the raw fields with more than 80% missing percentage were deleted (see Table 2).

Table 2. Original variables with more than 80% of data fields missing.

Name	Meaning	Percentage of Missing
code2	Merchant equipment id	98.8%
ip2, ip2_sub	Operating computer sign	90.7%
code1	Sub-stores id	90.6%
acc_id2, acc_id3	Account codes	89.1%
device_code3	Equipment id	89.0%
market_code	Marketing activity id	88.7%

User transaction flow data contain rich categorized information, large data volumes, and redundant information, which is not conducive to effective feature portraits of users. Therefore, it is necessary to feature mine the original transaction data to improve the accuracy of fraud detection. Feature mining is carried out in two dimensions: regular behavior and periodic features.

The conventional behavioral features include the number of user operations, the number of transactions, the transaction amount, the total transaction amount and frequency, etc. At the same time, in order to analyze the characteristics of the transaction amount, in addition to the quantitative features of this type in the summation, the mean, the variance, the quantile point, and other statistics, the amount of the transaction according to the numerical distribution of the value of the five transaction amount ranges was transformed into a new classification of features, as an important reference for mining behavioral features.

Cyclical features were mainly analyzed in the context of the actual situation; there are obvious cyclical features in the operation and transaction behavior of accounts in financial fraud scenarios, and large promotions often occur in a fixed number of days a week, so a large number of normal users and fraudulent users will be refluxed in the corresponding time nodes; thus, clustering the period of the account behavior is useful for mining the key features of the account. The transaction date was categorized into multiple period classes according to the distribution of weeks and transaction hours, and the specific distribution frequency of the categorized features was counted.

Finally, combined with the feature filtering method, the user node features were filtered and screened using variance to obtain 105-dimensional user features F.

4.1.2. Relationship Graph Acquisition

Node networking first obtained the set of four types of nodes (user, device, merchant, and address), and then obtained the directed edges of the three types of relationships to obtain the transaction heterogeneous network. For the four types of nodes, "UID" is the unique node identifier for users, "merchant" is the unique identifier for merchants, and the two types of nodes, equipment and address, involve multiple valid original fields as shown in Table 3.

Table 3. Description of valid raw fields of the nodes.

Node Type	Original Field	Explanation
Device	device1	Operating equipment parameter 1
	device2	Operating device parameter 2
	device_code1	Operating device unique identifier 1
	device_code2	Operation device unique identifier 2
Address	ip1	Operation IP address code
	ip1_sub	Operation IP address first three digits
	mac1	Operating MAC address code

To ensure that the relational network is accurately connected and to avoid excessive network sparsity, the two types of nodes, address and device, were compressed by defining the entity alignment means, which in return, realizes the compression of the number of

nodes. In the case of address nodes, the specific entity alignment means are shown in Figure 3. The address entity corresponding to ip1, ip1_sub, and mac1 were regarded as entities independent of each other for encoding, and multiple relationships can be obtained based on the original data; for the same user, when more than two address entities are the same, the combination of the address entity is considered to represent the same address. The new address entity is re-encoded and the operation is repeated to compress the device entity node to obtain the user transaction heterogeneous graph.

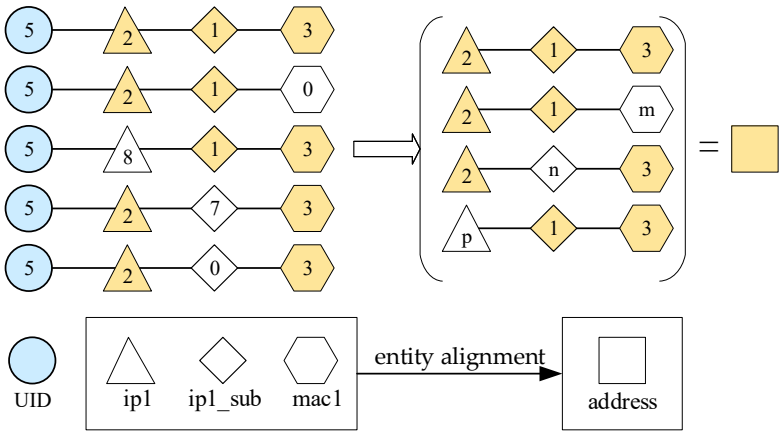


Figure 3. Schematic representation of the entity simplification process.

The above transaction heterogeneous network was randomly sampled to obtain four real datasets, which are described in Table 4, where U, D, M, and A denote the user, device, merchant, and address, respectively. Each dataset was split into a training set and a test set in a ratio of 7:3.

Table 4. Description of the dataset.

Dataset	(U, D, M, A)	Relationship	Number of Edges
Dataset 1	(10,393, 13,849, 8039, 2689)	U-D U-M U-A	14,835 23,038 12,995
Dataset 2	(10,393, 13,560, 8153, 2707)	U-D U-M U-A	14,529 23,334 12,965
Dataset 3	(10,393, 13,673, 8360, 2778)	U-D U-M U-A	14,652 23,444 13,158
Dataset 4	(15,590, 20,254, 11,478, 3273)	U-D U-M U-A	22,001 35,021 19,611

4.2. Experimental Environment and Parameter Settings

In this paper, the experimental hardware environment used was AMD R5-4600U with 16 GB of RAM, Windows 10 operating system, software version Python 3.7; the deep learning framework uses PyTorch 1.9.0, and the graph convolutional neural network framework uses DGL 0.9.1. DGL (Deep Graph), an open-source framework developed by New York University and Amazon.com., which seamlessly integrates with mainstream deep learning frameworks such as PyTorch, TensorFlow, etc. It provides multifunctional message-passing controls to simplify the process of building graph neural networks and provides speed optimization for batch computation and sparse matrices, which is good for graph computation with good scalability.

This model is divided into two layers; in the first layer, Tri-RGCN, the model hidden space dimension is set to 20, the input to 105 dimensions, and the output to 2 dimensions. The graph convolution layer setting routinely selects 2–3 layers; unlike general deep learning, performance can be improved by deepening the model, and a graph convolution neural network setting graph convolution layer that is too deep will lead to model degradation and over-smoothing phenomena [26]. The deeper reason is that the features will converge after constant aggregation and updating of the information of the central node; in addition, in terms of computational complexity, for the node classification task of aggregating the local information of neighboring nodes, a shallow graph convolution performance is better. In this paper, the RGCN model was set up with two layers of convolution, and the dichotomous graphs GUD, GUM, and GUA were input into the RGCN model, and the parameters of each RGCN model were updated with the back-propagation algorithm during the training process. The second layer is XGBoost integrated learning, and the three types of classification results obtained from the Tri-RGCN layer are used as the feature input which, combined with the XGBoost parallel learning decision-making characteristics, integrates the advantages of the three association modes in fraud detection. In the face of the discrete combinations of features, the XGBoost model, compared with the linear regression model and non-integrated tree model, shows better performance, but the computational complexity is low. In order to verify the classification performance of the model, the default parameters were chosen, and the integration effect was evaluated using a 50% cross-test.

4.3. Comparison Test

The selection of baseline models for the Tri- fraud detection performance test includes the first class of supervised machine learning models based on user features, such as LR (Logistic regression), KNN (K-Nearest Neighbor), and GBDT, and the second class of graph representations that take the whole graph as an object and aggregate three types of relationships at the same time, including GraphSage and RGCN. See Table 5 for details.

Table 5. Introduction to the baseline models.

Model	Model Description
LR [27]	Logistic regression introduces Sigmoid with conditional probability distribution as a decision function, which is a normalized linear regression model for continuous and discrete data.
KNN [28]	KNN obtains k proximity nodes by measuring the distance between different feature values and combines it with the majority voting method to achieve classification prediction.
GBDT [29]	Gradient Boosting Decision Trees are integrated learning methods that boost weak learners into strong learners by iteratively constructing decision trees to perform classification or regression tasks.
RGCN [12]	The GCN framework is applied to relational data to update the full graph node representation and classify the neighboring nodes with different edge types after linearly transforming the aggregated neighbor information.
GraphSage [30]	GraphSAGE provides an inductive framework to efficiently utilize the attribute information of nodes to learn node representations for local updating and classification after sampling neighboring node strategies.

The experimental comparison results are shown in Tables 6 and 7, where the bolded numbers are the optimal results for the specific metric. Firstly, we compare the first three classical supervised machine learning models with the commonly used graph representation learning methods, where the former directly inputs user features and the latter also needs to input the network structure. According to the experimental results, the GBDT tree model outperformed the LR and KNN algorithms, showing excellent performance in the face of nonlinear data and complex relationships. In contrast, the two graph representation

learning models had limited enhancement and no significant optimization compared to the tree model; the reason for this is that on the one hand, the GraphSage neighbor sampling, although it relieves the operational pressure, it may lead to the lack of important neighbor information of some nodes to a certain extent, and on the other hand, for the two models, the whole network structure of the full-graph input model contains four types of nodes and three types of relationships, and the increase in the number of relationships generates the problem of relationship overfitting, which affects the model performance.

Table 6. Comparison of dataset 1 and dataset 2.

Dataset	**Dataset 1**				**Dataset 2**			
Evaluation	**Acc**	**P**	**Recall**	**F1**	**Acc**	**P**	**Recall**	**F1**
LR	0.9171	0.8388	0.4800	0.6106	0.9129	0.7849	0.4986	0.6093
KNN	0.9219	0.7825	0.5903	0.6702	0.9221	0.7913	0.5964	0.6757
GBDT	0.9426	0.8948	0.6572	0.7557	0.9387	0.8905	0.6352	0.7393
RGCN	0.9427	0.8987	0.6500	0.7544	0.9499	0.9359	0.6795	0.7874
GraphSage	0.9319	0.8732	0.6074	0.7164	0.9358	0.9026	0.5943	0.7167
Detailed model	0.9541	0.8994	0.7461	0.8111	0.9551	0.9074	0.7479	0.8175

Table 7. Comparison of dataset 3 and dataset 4.

Dataset	**Dataset 3**				**Dataset 4**			
Evaluation	**Acc**	**P**	**Recall**	**F1**	**Acc**	**P**	**Recall**	**F1**
LR	0.9112	0.7969	0.4976	0.6107	0.9159	0.8264	0.4873	0.6123
KNN	0.9219	0.7917	0.6099	0.6863	0.9291	0.8180	0.6313	0.7092
GBDT	0.9405	0.9058	0.6476	0.7535	0.9409	0.8970	0.6462	0.7501
RGCN	0.9398	0.9063	0.6367	0.7480	0.9436	0.9202	0.6457	0.7589
GraphSage	0.9319	0.9042	0.5757	0.7035	0.9350	0.8727	0.6150	0.7215
Detailed model	0.9525	0.9072	0.7402	0.8120	0.9552	0.9162	0.7439	0.8178

Comparing and analyzing this paper's model with the baseline method in Table 6, in dataset 1 and dataset 2, this paper's model outperformed the baseline model, with the most obvious increase in Recall and F1. Taking the better-performing GBDT tree model as a reference, the Recall and F1 metrics in dataset 1 increased by 13.5% and 7%, respectively, and dataset 2 increased by 17.7% and 10.5%. See Table 7 for an experimental comparison of the performance of this paper's model with dataset 3 and dataset 4 in terms of the various metrics which mostly outperformed the baseline model. Compared to the RGCN model with full graph input, the recall of this paper's model using dataset 3 and dataset 4 improved by more than 10%. Analyzing the experimental results on the four datasets, this paper's model demonstrates some optimization in each evaluation index, proving the detection performance of the model. Meanwhile, combined with the analysis of real business scenarios, this paper's model had a significantly improved recall rate, which is of practical significance for reducing normal user misjudgment.

4.4. Ablation Experiment

The main innovation of the model in this paper compared to previous work is the disambiguation of the multiple relationships with decision-side fusion. Therefore we validate the effectiveness of the model by comparing the direct input graph convolutional neural network model for multiple relational views and comparing the model for decision fusion; the results show that our model performed better. Firstly, see Tables 4 and 5 to compare the direct input of the three relational views into the RGCN model, which proves that there was a decrease in the performance when directly utilizing multiple relational classifications.

Figure 4 shows that, with the four datasets, the XGBoost model performed optimally in decision fusion of the classification results after inputting the three relational views

into the RGCN model separately, further proving that the XGBoost model has a better performance in coping with decision fusion of multiple relationships.

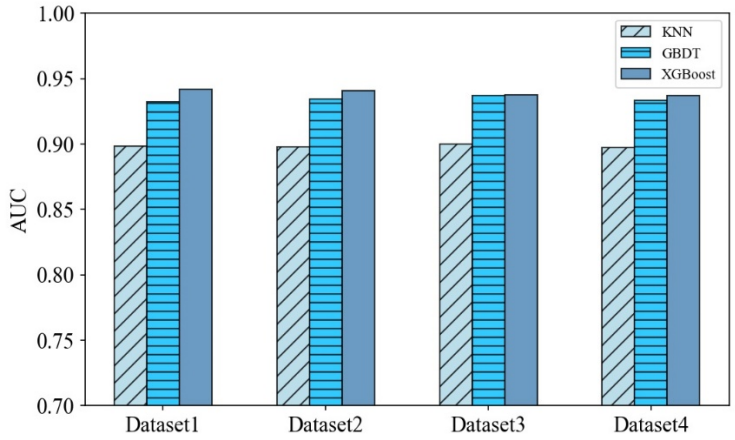
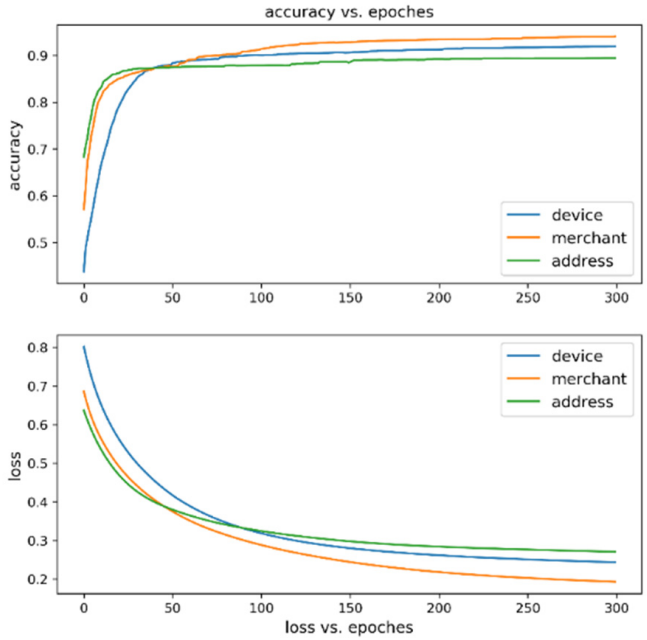


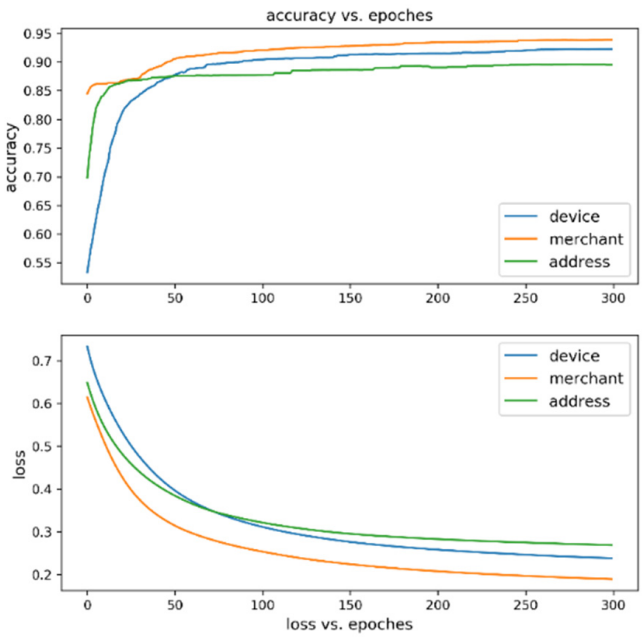
Figure 4. Comparison of decision fusion models.

4.5. Comparison of Behavioral Patterns

The bipartite graphs GUD, GUM, and GUA correspond to three association situations, i.e., three behavioral patterns for user–device, user–merchant, and user–address interactions, respectively. In order to study the optimization process of the Tri-RGCN layer and the fraud detection effect of each behavioral pattern, the bipartite graph performance of the RGCN layer was compared and analyzed. The change in accuracy and loss during the RGCN optimization process on datasets 1–4 is shown in Figure 5.

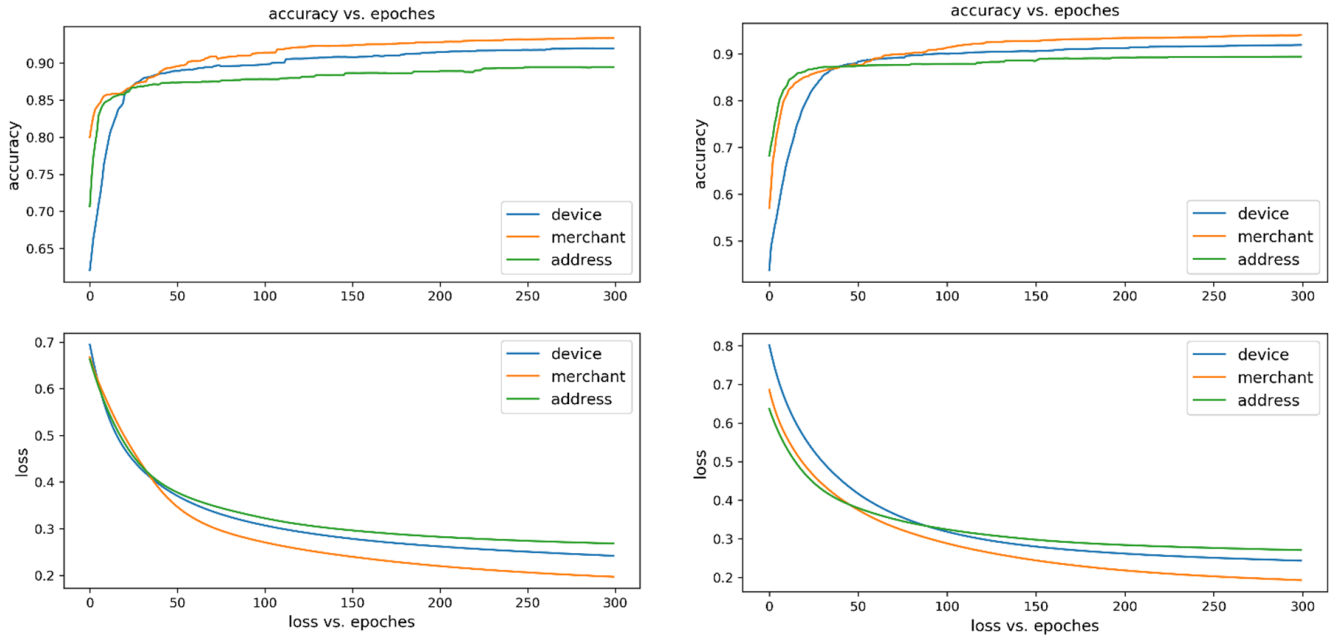


(**a**) Dataset 1



(**b**) Dataset 2

Figure 5. *Cont.*



(**c**) Dataset 3 (**d**) Dataset 4

Figure 5. Optimization process.

From Figure 5, it can be seen that along with the increase in the number of epochs, the accuracy of fraud detection was effectively improved, the loss was gradually reduced and stabilized, and the prediction performance of each bipartite graph input into the RGCN model was improved. This indicates that all three behavioral patterns have improved account node fraud detection performance, and in the four datasets, the device–merchant GUM bipartite graph classification was the best, that is, through the merchant associated with the user's behavioral pattern, the abnormal node identification effect of this data is more significant and effective.

To further validate the performance of different behavioral patterns in the model, the Shap value is introduced to rank the feature importance of the XGBoost layer. Shap (Shapley additive explanations) originated from cooperative game theory and is often used to mine important nonlinear relationships in models and explain complex machine learning models [31]. Using the XGBoost layer of the model in this paper, with the prediction results of bipartite graph convolutional classification of three behavioral patterns as inputs, the shape value feature ranking performance on each dataset is shown in Figure 6. The feature correspondences are as follows:

$$f_{RGCN}(GUD) \rightarrow (F1, F2)$$

$$f_{RGCN}(GUM) \rightarrow (F3, F4)$$

$$f_{RGCN}(GUA) \rightarrow (F5, F6)$$

It can be seen that in the four datasets, the feature "F4" showed a strong correlation with the prediction results, which further verifies that the user–merchant behavioral pattern has a high reference value in the process of mining fraudulent users. Relatively speaking, the other two behavioral patterns in the experimental datasets showed a lower degree of importance, but combined with the accuracy of the results of the graph convolution layer output and other indicators, it also shows that the construction of an effective relational network on the performance of fraud detection is very obvious.

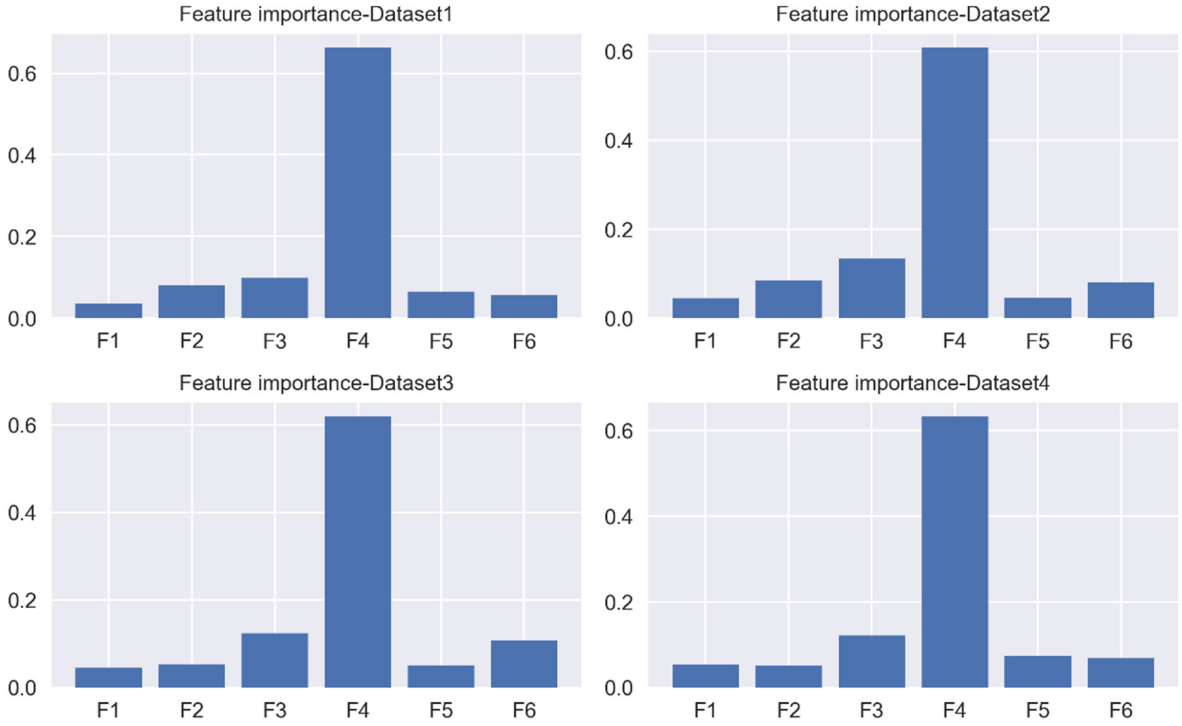


Figure 6. Feature importance.

5. Conclusions

In this paper, a multiple relational graph fraud detection model Tri-RGCN-XGBoost that integrates multiple behavioral patterns was designed. The main idea lies in mining various types of user associations in the fraudulent scenario, using graph convolutional neural network as the initial framework and aggregating the user's neighboring node information under multiple behavioral associations. Finally, the final fraud prediction is obtained by fusing the classification results under the three relational graphs by XGBoost. In the experimental validation and comparative analysis on real datasets, the model in this paper outperformed the baseline model in each evaluation index, proving the feasibility and effectiveness of the model. Combining the single bipartite graph convolution iteration with the XGBoost layer feature importance ranking verified the improvement of the fraud detection effect of the experimental data in this paper, and further explains the integrated model.

The method proposed in this paper shows excellent performance in fraud detection but is limited by the fact that the data only focus on three behavioral patterns and networked experiments. In addition, the process does not consider the edge attributes of the heterogeneous graph, which simplifies the node information aggregation function; thus, the next step should focus on the reasonable mining and application of the edge attributes to improve the fraud detection effect based on the heterogeneous graph.

Author Contributions: Conceptualization, J.L.; methodology, J.L.; software, D.Y.; validation, J.L.; formal analysis, D.Y.; investigation, J.L. and D.Y.; resources, J.L. and D.Y.; data curation, J.L. and D.Y.; writing—original draft preparation, J.L. and D.Y.; writing—review and editing, J.L. and D.Y.; visualization, J.L. and D.Y..; supervision, J.L.; project administration, D.Y.; funding acquisition, J.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Available on request.

Conflicts of Interest: The authors declare that they have no competing interest.

References

1. Sun, Q.; Tang, T.; Zheng, J.B.; Pan, Q.; Zhao, J.T. Financial Transaction Data Based Intelligent FraudGraph Network Detection. *J. Appl. Sci.-Electron. Inf. Eng.* **2020**, *38*, 713–723.
2. Du, H.; Li, D.; Wang, W. Abnormal User Detection via Multiview Graph Clustering in the Mobile e-Commerce Network. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 3766810. [CrossRef]
3. Liu, H.L.; Liu, Y.X.; Xu, J.Y.; Chen, S.H.; Qiao, L. Research progress in the application of graph anomaly detection in financial anti-fraud. *Comput. Eng. Appl.* **2022**, *58*, 41–53.
4. Liu, J.; Shang, X.Q.; Song, L.Y.; Tan, Y.C. Progress of Graph Neural Networks on Complex Graph Mining. *J. Softw.* **2022**, *33*, 3582–3618.
5. Zhang, G.; Wu, J.; Yang, J.; Beheshti, A.; Xue, S.; Zhou, C.; Sheng, Q.Z. FRAUDRE: Fraud detection dual-resistant to graph inconsistency and imbalance. In Proceedings of the 2021 IEEE International Conference on Data Mining (ICDM), Auckland, New Zealand, 7–10 December 2021; pp. 867–876.
6. Zhang, W.; Liu, X.; Zhang, X.; Hu, W.; Zhang, J.; Shao, W. Medicare Fraud Gang Discovery Based on Community Discovery Algorithms. In Proceedings of the 2022 IEEE 8th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), Jinan, China, 6–8 May 2022; pp. 206–211.
7. Liu, F.; Li, Z.; Wang, B.; Wu, J.; Yang, J.; Huang, J.; Zhang, Y.; Wang, W.; Xue, S.; Nepal, S. eRiskCom: An e-commerce risky community detection platform. *VLDB J.* **2022**, *31*, 1085–1101. [CrossRef]
8. Mo, C.; Li, S.; Tso, G.K.; Zhou, J.; Qi, Y.; Zhu, M. Motif-aware temporal GCN for fraud detection in signed cryptocurrency trust networks. *arXiv* **2022**, arXiv:2211.13123.
9. Yu, T.; Chen, X.; Xu, Z.; Xu, J. MP-GCN: A Phishing Nodes Detection Approach via Graph Convolution Network for Ethereum. *Appl. Sci.* **2022**, *12*, 7294. [CrossRef]
10. Jing, R.; Tian, H.; Zhou, G.; Zhang, X.; Zheng, X.; Zeng, D.D. A GNN-based Few-shot learning model on Credit Card Fraud detection. In Proceedings of the 2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI), Beijing, China, 15 July–15 August 2021; pp. 320–323.
11. Zhu, J.; Wang, Q.; Tao, C.; Deng, H.; Zhao, L.; Li, H. AST-GCN: Attribute-augmented spatiotemporal graph convolutional network for traffic forecasting. *IEEE Access* **2021**, *9*, 35973–35983. [CrossRef]
12. Kanezashi, H.; Suzumura, T.; Liu, X.; Hirofuchi, T. Ethereum Fraud Detection with Heterogeneous Graph Neural Networks. *arXiv* **2022**, arXiv:2203.12363.
13. Deng, Z.; Xin, G.; Liu, Y.; Wang, W.; Wang, B. Contrastive graph neural network-based camouflaged fraud detector. *Inf. Sci.* **2022**, *618*, 39–52. [CrossRef]
14. Jiang, N.; Duan, F.; Chen, H.; Huang, W.; Liu, X. MAFI: GNN-based multiple aggregators and feature interactions network for fraud detection over heterogeneous graph. *IEEE Trans. Big Data* **2021**, *8*, 905–919. [CrossRef]
15. Zhang, G.; Li, Z.; Huang, J.; Wu, J.; Zhou, C.; Yang, J.; Gao, J. eFraudcom: An e-commerce fraud detection system via competitive graph neural networks. *ACM Trans. Inf. Syst. (TOIS)* **2022**, *40*, 1–29. [CrossRef]
16. Tan, X.Y.; Pei, S.W. Research on Heterogeneous Graph Neural Network Based on Higher-Order Neighbors. *J. Chin. Comput. Syst.* **2023**, *44*, 1954–1960.
17. Li, Q.; He, Y.; Xu, C.; Wu, F.; Gao, J.; Li, Z. Dual-Augment Graph Neural Network for Fraud Detection. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, 17–21 October 2022; pp. 4188–4192.
18. Liu, Y.; Ao, X.; Qin, Z.; Chi, J.; Feng, J.; Yang, H.; He, Q. Pick and choose: A GNN-based imbalanced learning approach for fraud detection. *Proc. Web Conf.* **2021**, *2021*, 3168–3177.
19. Zhang, Z.; Dong, Y.; Wu, H.; Song, H.; Deng, S.; Chen, Y. Metapath and syntax-aware heterogeneous subgraph neural networks for spam review detection. *Appl. Soft Comput.* **2022**, *128*, 109438. [CrossRef]
20. Hassanzadeh, R.; Nayak, R.; Stebila, D. Analyzing the effectiveness of graph metrics for anomaly detection in online social networks. In Proceedings of the Web Information Systems Engineering-WISE 2012: 13th International Conference, Paphos, Cyprus, 28–30 November 2012; pp. 624–630.
21. Fu, X.L.; Yan, C.W.; Song, M.Q.; Xu, S.Q. Graph Representation Learning Based Group Fraud Risk Detection in the Consumer Finance. *J. Chin. Inf. Process.* **2022**, *36*, 120–128+138.
22. Moradi, F.; Olovsson, T.; Tsigas, P. Overlapping communities for identifying misbehavior in network communications. In Proceedings of the Advances in Knowledge Discovery and Data Mining: 18th Pacific-Asia Conference, PAKDD 2014, Tainan, Taiwan, 13–16 May 2014; pp. 398–409.
23. Qi, A.J.; Fan, X.; Dong, X.J.; Chu, Y.J.; Yuan, X.R. Community discovery based network anomaly detection method. *Chin. J. Comput.* **2022**, *45*, 825–837.
24. Huang, X.; Yang, Y.; Wang, Y.; Wang, C.; Zhang, Z.; Xu, J.; Chen, L.; Vazirgiannis, M. Dgraph: A large-scale financial dataset for graph anomaly detection. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 22765–22777.
25. Niu, X.J.; Ling, F. Study on Personal Credit Risk AssessmentModel Based on Hybrid Learning. *J. Fudan Univ. Nat. Sci.* **2021**, *60*, 703–719.

26. Zhang, W.; Sheng, Z.; Jiang, Y.; Xia, Y.; Gao, J.; Yang, Z.; Cui, B. Evaluating deep graph neural networks. *arXiv* **2021**, arXiv:2108.00955.
27. Wang, T.; Zhao, Y. Credit Card Fraud Detection using Logistic Regression. In Proceedings of the 2022 International Conference on Big Data, Information and Computer Network (BDICN), Sanya, China, 20–22 January 2022; pp. 301–305.
28. Sisodia, D.; Sisodia, D.S. Quad division prototype selection-based k-nearest neighbor classifier for click fraud detection from highly skewed user click dataset. *Eng. Sci. Technol. Int. J.* **2022**, *28*, 101011. [CrossRef]
29. Hancock, J.T.; Khoshgoftaar, T.M. Gradient boosted decision tree algorithms for medicare fraud detection. *SN Comput. Sci.* **2021**, *2*, 268. [CrossRef]
30. Van Belle, R.; Van Damme, C.; Tytgat, H.; De Weerdt, J. Inductive graph representation learning for fraud detection. *Expert Syst. Appl.* **2022**, *193*, 116463. [CrossRef]
31. Yan, C.; Chi, X.Y.; Liu, X.H. The application of XGBoost and SHAP to examining the factors in freight truck-related crashes: An exploratory analysis. *Accid. Anal. Prev.* **2022**, *39*, 168–173.